REMARKS

Claims 1-7, 9-19, 21, 22, 24-28, 31 and 33-39 are currently pending in the subject application and are presently under consideration. Claims 1, 15, 21, 31 and 33 have been amended as shown on pages 2-8 of the Reply.

Favorable reconsideration of the subject patent application is respectfully requested in view of the comments and amendments herein.

I. Rejection of Claims 1-7, 9-19, 21-22, and 24-28 Under 35 U.S.C §112

Claims 1-7, 9-19, 21-22, and 24-28 stand rejected under 35 U.S.C. §112, second paragraph, due to certain informalities. Withdrawal of this rejection is requested in view of amendments to independent claims 1 and 21.

II. Rejection of Claims 1-7, 9-12, 14-19, and 33-39 Under 35 U.S.C. §103(a)

Claims 1-7, 9-12, 14-19, and 33-39 stand rejected under 35 U.S.C. §103(a) over Crater *et al.* (US 6,201,996-hereinafter Crater) in view of Muller *et al.* (US 6,480,489-hereinafter Muller). This rejection should be withdrawn for at least the following reason. Crater and Muller, either alone or in combination, do not teach or suggest each and every aspect of the claimed subject matter.

The claimed subject matter relates to facilitating optimized data transfers between an industrial controller and one or more remote client applications by mitigating the amount of information communicated across the network to the PLC. A client machine queries the industrial controller, which along with returning data items in accordance with the query, an aggregation component is created and installed on the industrial controller. The aggregation component creates one or more optimized packets, the one or more optimized packets store data items associated with the query. Further, the aggregation component monitors the industrial controller and any new data that, if it pertains to the query, combines it with the data currently stored in an optimized packet. Depending upon various requirements, such as network data transmission restrictions, the aggregation component can dynamically adjust the size of the optimized packet.

In particular, independent claim 1, as amended, recites in part:... *the primary aggregation component is created and defined in response to a query received from an entity*

remote to the industrial controller and is installed on the industrial controller, ***the primary aggregation component aggregates one or more selected data items into an aggregated subset of data items according to a memory address of a first data item in a group, followed by a length and then followed by values relating to the data items in the group****; a communications component associated with the remote entity, the communications component transmits the aggregated subset of data items via a singular communications packet across a network and adds at least one secondary aggregation component at the industrial controller based upon at least one of increased data demands and network protocol considerations; and a component associated with the remote entity, the component receives handle information from the industrial controller relating to the selected data items and employs only the handle information as a reference with consistent length to generate an update data packet to update data locations in the industrial controller.* Crater and Muller do not teach or suggest such aspects of the claimed subject matter.

Crater relates to communicating among programmable controllers for operating and monitoring industrial processes and equipment. Crater provides an object-oriented control structure that facilitates communication between an industrial controller and a remote computer. The control structure is organized around a database of object items each associated with a control function. For each control function, the items include one or more procedures for performing an action associated with the control function.

Muller is presented to overcome the deficiencies of Crater in failing to disclose aggregating one or more data items into an aggregated subset of data items. Muller relates to a system and method for transferring a packet received from a network to a host computer according to an operation code associated with the packet. Based on some of the retrieved information, a transfer engine stores the packet in one or more host memory buffers. If the packet was formatted with one of the set of predetermined protocols, its data is re-assembled in a re-assembly buffer with data from other packets in the same communication flow and re-assembled data is provided to a destination application or user. However, Muller is silent with regard to ***the primary aggregation component aggregates one or more selected data items into an aggregated subset of data items according to a memory address of a first data item in a group, followed by a length and then followed by values relating to the data items in the group.***

Muller provides for identifying related packets (packets that are part of one flow) by their flow numbers or flow keys and transferring data from the related packets together (*see*, Col. 55, lines 15-20). One or more headers of an incoming packet are examined or parsed (*e.g.*, headers for the layer two, three and four protocols) in incoming network traffic in order to identify the packet's source and destination entities. Using identifiers of the communicating entities as a key, data from multiple packets may be aggregated or re-assembled for a pair of communicating entities. Typically, a datagram sent to one destination entity from one source entity is transmitted via multiple packets. Aggregating data from multiple related packets (*e.g.*, packets carrying data from the same datagram) allows a datagram to be re-assembled and collectively transferred to a host computer or to the destination entity in a highly efficient manner (*see*, Col. 8, lines 15-30). A datagram is defined as a collection of data sent from one entity to another and comprises data transmitted in multiple packets for same sending and receiving entity (*see*, Col. 14, lines 35-40). The re-assembly of data involves the re-assembly or combination of data from multiple related packets (*i.e.* packets from a single communication flow or a single datagram) (*see*, Col. 35, lines 64-67). Hence, Muller provides for identifying and gathering multiple packets for a source entity and destination entity pair and sending those multiple packets collectively, instead of sending them serially. More particularly, Muller provides for *gathering the data from the multiple packets for a source entity and destination entity pairing*. Hence each pair of sending and receiving entities is identified for combining data packets belonging to the same pair of sending and receiving entities. However, Muller does not contemplate aggregating one or more selected data items in a data packet for a sending and receiving entity into an aggregated subset of data items ***according to a memory address of a first item in a group, followed by a length and then followed by the values relating to the items in the group***. Further Muller provides for re-assembling only packets that are formatted in accordance with one or more of a set of pre-selected protocols (*see*, Col. 4, lines 35-40). However nowhere does Muller teach or suggest ***aggregating one or more selected data items into an aggregated subset of data items according to a memory address of a first item in a group, followed by a length and then followed by the values relating to the items in the group***. This feature facilitates mitigating repeated and redundant header and ending data that is generally associated with a network communications packet. *A plurality of related or unrelated* (not in contiguous memory portions or of the same data type) *data items are aggregated* and transmitted *in a singular*

communications packet, thereby mitigating overhead associated with transmitting these items according to individual data item requests.

Further, at page 8 of the subject Final Office Action, it is contended that Muller teaches *the component receives handle information from the industrial controller relating to the selected data items and employs only the handle information as a reference with consistent length to generate an update data packet to update data locations in the industrial controller*, with respect to independent claim 1. Muller provides for identifying empty buffers into which packets are to be stored via a free descriptor ring that is maintained in host memory. A descriptor ring contains descriptors including data, flag, pointer and address for storing information. Each descriptor including data, flag, pointer and address stores its index within the free descriptor ring and an identifier including memory address and pointer of a free buffer that is used to store packets. *The buffer is identified in a descriptor by its address in memory* (*see*, Col. 55, lines 25-67 & Col. 56, lines 1-45). When a packet is stored in a buffer, *a complete descriptor in a complete descriptor ring is configured to convey relevant information concerning the packet to the host computer*. The complete descriptor stores header index, to identify the buffer that contains a header portion of the packet and a data index to identify the buffer that contains a data portion of the packet (*see*, Col. 5, lines 1-29). Hence, Muller provides for only identifying empty buffers into which packets are to be stored. The buffer is identified in a descriptor by its address in memory and the descriptors include data, flag, pointer and address. More particularly, Muller provides for employing complete memory address of the memory to identify empty buffer and storing data packets, wherein memory address is specified by descriptors including data, flag, pointer and address for storing information. However, Muller does not contemplate ***employing only the handle information as a reference with consistent length to generate an update data packet to update data locations in the industrial controller***. The handle information is similar to *an indirect address indication of the location of the requested data item in the controller*. This feature facilitates conserving the network bandwidth by employing data type pointers or handles *rather than explicit name identifiers* or complete memory address (as employed in the system provided by Muller) as part of an update header associated with an update request. This mitigates the need to use storage locations, pointers and explicit tags that are often lengthy and consist of variable lengths thereby causing variable and often larger amounts of data to be transmitted. The handle is employed as a consistent one or two byte data reference (or other

consistent amount) that generally tends to mitigate the overall amount of data to be transmitted when compared to explicit tag references. The handle provides an indirect indication having fixed length (*e.g.,* handles providing 2 byte pointer as opposed to variable length explicit tag names), thus mitigating the amount of information communicated across the network to the PLC when indicating which data item is to be altered.

Further, independent claim 33, as amended, recites in part:...***a primary aggregation component that aggregates one or more selected data items into an aggregated subset of data items according to a memory address of a first data item in a group, followed by a length and then followed by values relating to the data items in the group, the primary aggregation component defined and installed at the industrial controller by an entity remote from the industrial controller.*** As discussed above, Crater and Muller, do not teach or suggest the aggregation of the data subsets according to a memory address of the first data item in a group.

In view of at least the foregoing, Crater and Muller, either alone, or in combination, do not teach or suggest each and every feature as recited in independent claims 1 and 33 (and claims 2-7, 9-12, 14-19, and 34-39 that depend respectively therefrom). Accordingly it is believed that the subject claims are in condition for allowance, and withdrawal of this rejection is respectfully requested.

III. Rejection of Claim 13 Under 35 U.S.C. §103(a)

Claim 13 stands rejected under 35 U.S.C. §103(a) over Crater *et al.* (US 6,201,996-hereinafter Crater) and Muller *et al.* (US 6,480,489-hereinafter Muller) in view of Bhatt *et al.* (US 6,097,399-hereinafter Bhatt). This rejection should be withdrawn for at least the following reason. The subject claims depend from independent claim 1, and as discussed *supra*, Crater and Muller do not teach or suggest all aspects of amended independent claim 1; and Bhatt. does not make up for the deficiencies of Crater and Muller. Therefore, it is respectfully requested that this rejection be withdrawn.

IV. Rejection of Claims 21, 22, 24, and 31 Under 35 U.S.C. §103(a)

Claims 21, 22, 24, and 31 stand rejected under 35 U.S.C. §103(a) over Muller *et al.* (US 6,480,489-hereinafter Muller) in view of Crater *et al.* (US 6,201,996 –hereinafter Crater). This

rejection should be withdrawn for at least the following reasons. The cited references, either alone or in combination, do not teach or suggest all aspects of the subject claims.

The claimed subject matter relates to facilitating optimized data transfers between an industrial controller and one or more remote client applications by mitigating the amount of information communicated across the network to the PLC.

In particular, independent claim 21, as amended, recites in part:... ***the data items arranged according to at least one of contiguous or non-contiguous address memory locations and receiving data from the object that has been updated by the controller, receiving handle information from the industrial controller relating to the selected data items and employing only the handle information as a reference with consistent length to generate an update data packet to update data locations in the industrial controller.*** Muller and Crater do not teach or suggest such aspects.

Muller relates to a system and method are provided for transferring a packet received from a network to a host computer according to an operation code associated with the packet. Based on some of the retrieved information, a transfer engine stores the packet in one or more host memory buffers. If the packet was formatted with one of the set of predetermined protocols, its data is re-assembled in a re-assembly buffer with data from other packets in the same communication flow and re-assembled data is provided to a destination application or user.

Crater relates to communicating among programmable controllers operating and monitoring industrial processes and equipment. Crater provides an object-oriented control structure that facilitates communication between an industrial controller and a remote computer. The control structure is organized around a database of object items each associated with a control function. For each control function, the items include one or more procedures for performing an action associated with the control function; and this reference does not teach the claimed features.

Muller provides for using different types of host memory areas (or buffers) for storing different types of packets. A re-assembly buffer is used to re-assemble data from multiple packets of a single communication flow. Collecting multiple data portions in a single buffer allows efficient transfer of the data to a destination application or user. A packet eligible for re-assembly is stored across two buffers, its data in a re-assembly buffer and its header in a header buffer (*see*, Col. 4, lines 45-57). Using identifiers of the communicating entities as a key, data

from multiple packets are aggregated or re-assembled. Typically, a datagram sent to one destination entity from one source entity is transmitted via multiple packets. Aggregating data from multiple related packets (*e.g.*, packets carrying data from the same datagram) thus allows a datagram to be re-assembled and collectively transferred to a host computer. The datagram is then provided to the destination entity in a highly efficient manner. For example, rather than providing data from one packet at a time in separate "copy" operations, an entire memory page of data is provided to the destination entity, possibly in exchange for an empty page (*see*, Col. 8, lines 20-37). Hence, Muller only provides for re-assembling or aggregating data from multiple related data packets for a source entity and destination entity pair and collectively transferring the data to the destination entity instead of transferring data through multiple packets. It is respectfully submitted that the data packets after re-assembly are stored across two buffers, the data in a re-assembly buffer and the header in a header buffer. Essentially, Muller provides for assembling data for multiple packets in one buffer and headers for the multiple packets in another buffer. However, Muller does not contemplate ***adding data items of interest to the object, the data items arranged according to at least one of contiguous or non-contiguous address memory locations*** and thus mitigating the amount of overhead associated with transferring data items as individual or unrelated entities.

Further, independent claim 31, as amended, recites in part:*...means for requesting, by the processor, tag identifiers from a controller; means for constructing an optimized data packet from the tag identifiers requested from the controller; means for installing the optimized data packet on the controller;* ***means for refreshing the optimized data packet on the controller; means for adding data items of interest to the data packet, the data items arranged according to at least one of contiguous or non-contiguous address memory locations;*** *means for transmitting data from the optimized data packet that has been refreshed by the controller; and means for updating the controller via employment of handle information as a reference with consistent length.* As discussed above, Muller and Crater do not teach or suggest adding data items of interest to the object, the data items arranged according to at least one of contiguous or non-contiguous address memory locations and thus mitigating the amount of overhead associated with transferring data items as individual or unrelated entities.

In view of at least the foregoing, it is respectfully submitted that Muller and Crater, either alone, or in combination, do not teach or suggest each and every element as recited in

independent claims 21 and 31 (and dependent claims 22 and 24 that depend from independent claim 21). Accordingly, it is believed that the subject claims are in condition for allowance, and withdrawal of this rejection is respectfully requested.

V. Rejection of Claims 25-26 Under 35 U.S.C. §103(a)

Claims 25-26 stand rejected under 35 U.S.C. §103(a) over Muller *et al.* (US 6,480,489-hereinafter Muller) and Crater *et al.* (US 6,201,996-hereinafter Crater) in view of Patel (US 6,889,257-hereinafter Patel). Withdrawal of this rejection is requested for at least the following reason. As discussed *supra* with regard to independent claim 21, the cited references Muller and Crater, individually or in combination, do not teach or suggest all aspects recited in the subject claims. Patel does not make up for the deficiencies of Muller and Crater with respect to independent claim 21 (from which claims 25 and 26 depend from). Hence, it is respectfully submitted that this rejection be withdrawn.

VI. Rejection of Claims 27-28 Under 35 U.S.C. §103(a)

Claims 27-28 stand rejected under 35 U.S.C. §103(a) over Muller *et al.* (US 6,480,489-hereinafter Muller) and Crater *et al.* (US 6,201,996-hereinafter Crater) in view of McCoskey *et al.* (US 2003/0028889-hereinafter McCoskey). This rejection should be withdrawn for at least the following reason. The cited references, either alone or in combination, do not teach or suggest all aspects of the subject claims. As discussed *supra* with regard to independent claim 21, the cited references Muller and Crater, individually or in combination, do not teach or suggest all aspects recited in the subject claims. McCoskey does not make up for the deficiencies of Muller and Crater with respect to independent claim 21 (from which claims 27 and 28 depend from). Thus, it is respectfully requested that this rejection be withdrawn.

CONCLUSION

The present application is believed to be in condition for allowance in view of the above comments and amendments. A prompt action to such end is earnestly solicited.

In the event any fees are due in connection with this document, the Commissioner is authorized to charge those fees to Deposit Account No. 50-1063 [ALBRP284US].

Should the Examiner believe a telephone interview would be helpful to expedite favorable prosecution, the Examiner is invited to contact applicants' undersigned representative at the telephone number below.

Respectfully submitted,

TUROCY & WATSON, LLP

/Thomas E. Watson/
Thomas E. Watson
Reg. No. 43,243

TUROCY & WATSON, LLP
57th Floor, Key Tower
127 Public Square
Cleveland, Ohio 44114
Telephone (216) 696-8730
Facsimile (216) 696-8731